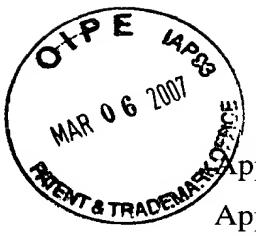
IFW
AF



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
Before the Board of Patent Appeals and Interferences

Applicants	:	Mehmet Kemal Ozkan et al.
Application No.:		09/622,331
Filed	:	March 19, 2001
For	:	A SYSTEM FOR FORMING, PARTITIONING AND PROCESSING PROGRAM GUIDES
Examiner	:	Hai V. Tran
Art Unit	:	2623

APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

May It Please The Honorable Board:

Applicants appeal from the FINAL Office Action dated October 19, 2006, in which claims 1-16 of the above-identified application stand rejected.

Applicants waive an Oral Hearing for this appeal.

Please charge the $500.00 fee for filing this Brief to Deposit Account No. 07-0832.

Please charge any additional fee or credit overpayment to the above-indicated Deposit Account No. 07-0832.

03/07/2007 RMEBRAHT 00000033 070832 09622331
01 FC:1402 500.00 DA

****

Certificate of Mailing under 37 CFR 1.8

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in a postage paid envelope addressed to: Mail Stop: Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date indicated below.

Signature Lori Kleeman Date: March 2, 2007

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

The real party in interest of Application No. 09/622,331 is:

THOMSON LICENSING
46, Quai A. Le Gallo
F-92100 Boulogne-Billancourt
France.

II. RELATED APPEALS AND INTERFERENCES

There are no related Appeals or Interferences.

III. STATUS OF THE CLAIMS

Claims 1-16 are pending in this application. Claims 17 and 18 have been canceled.

Claims 1-16 have been rejected.

The rejection of claims 1-16 are appealed.

IV. STATUS OF AMENDMENTS

There have been no amendments after the FINAL Office Action dated October 19, 2006.

In response to the FINAL Office Action dated October 19, 2006, Applicants' representative filed a Notice of Appeal on February 1, 2007, along with a petition for a one month extension of time.

This appeal is directed to the claims as they stood at the time of the FINAL Office Action of October 19, 2006, which are shown in the Claims Appendix of this Brief.

V. SUMMARY OF CLAIMED SUBJECT MATTER

There are four independent claims in the application: 1, 6, 8 and 13.

Applicants' independent claims are directed to forming, partitioning and processing of program guides. (Applicants' specification, Title; and p. 2, lns. 24-34.) Program guides comprise a number of tables. (Applicants' specification, p. 2, lns. 24-34; FIGs. 5, 6, and 7.)

In this regard, independent claim 1 is directed to an apparatus that detects changes in tables of a program guide by the use of specific types of version information. In particular, claim 1 is directed to an apparatus that uses a first version identifier to determine if something has changed in the program guide and, if something has changed, uses a second version identifier to determine what particular table has changed. The second version identifier is updated in response to a version change in a tertiary table hierarchically linked to the second table. (Claim 1, lns. 14-16; Applicants' specification, p. 2, lns. 26-29; p. 14, ln. 32 to p. 15, ln. 28; FIGs. 5, 6 and 7.)

Turning next to independent claim 6, this claim is directed to an apparatus that processes a program guide having individual partitions, where each partition is assigned to a partition identifier and wherein the partitions are dynamically re-partitionable by re-assignment of the partition identifiers. (Claim 6, lns. 3-8; p. 6, ln. 37 to p. 7, ln. 23; p. 11, lns. 20-29 and FIGs. 1, 9 and 10; p. 11, ln. 36 to p. 12, ln. 20 and FIGs. 4 and 8.)

Finally, independent claims 8 and 13 are complementary to claims 1 and 6 in that claims 8 and 13 are directed to methods of forming the packetized program data. In particular, claim 8 is directed to a method for forming packetized program data for processing in a decoder, where the packetized program data includes a first version identifier and a second version identifier. (Claim 8, lns. 5-12; Applicants' specification, p. 16, lns. 28-33; p. 17, lns. 18-20; FIG. 12.) With respect to claim 13, this claim is directed to a method for forming packetized program data for processing in a decoder, where the packetized program data includes updatable version numbers for indicating content change of a partition and cell numbers that enable dynamic re-partitioning of program guide information. (Claim 13, lns. 3-12; Applicants' specification, p. 16, ln. 28 to p. 17, ln. 29; FIG. 12.)

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Whether claims 1 to 16 are unpatentable under 35 U.S.C. § 103(a) over U.S. Patent No. 6,160,545 issued December 12, 2000 to Eyer et al. (*Eyer*) in view of the "Program and System Information Protocol for Terrestrial Broadcast and Cable ", ATSC Standard, Doc. A/65 dated December 23, 1997 (*PSIP*).

VII. ARGUMENT

Rejection of Claims 1-16 under 35 U.S.C. § 103(a) as being unpatentable over *Eyer* in view of *PSIP*

For the purposes of this appeal only, and without prejudice, Applicants will agree with the Examiner's position on *Eyer*. However, Applicants respectfully disagree with the Examiner's interpretation of *PSIP*. In particular, the combination of *Eyer* and *PSIP* does not yield Applicants' claimed invention.

CLAIMS 1-5, 8-12

CLAIMS 1-5 ARE PATENTABLE

Applicants' independent claim 1 requires at least

> (1) a tertiary table hierarchically linked to said secondary table; and
>
> (2) a second version identifier conveyed in a secondary data table and updated in response to ... a version change in the tertiary table.
>
> Claim 1, lns. 12-13; emphasis added.

Nowhere does either *Eyer*, or *PSIP* – singly or in combination – describe or suggest a tertiary table **hierarchically linked** to the secondary table as claimed by Applicants. In addition, nowhere does either *Eyer*, or *PSIP* – singly or in combination – describe or suggest **a second version identifier conveyed in a secondary data table and updated in response to ... a version change in the tertiary table.**

Background information on PSIP

However, before getting to the Examiner's argument, some background information on *PSIP* must be provided. *PSIP* is a standard that defines a Program and System Information Protocol for providing information about television (TV) programming to a TV receiver, e.g., a TV set. Since there are different types of information, *PSIP* defines different tables to convey different information. *PSIP*, p. 1. Of interest here are three *PSIP* tables: a Master Guide Table (MGT), an Event Information Table (EIT) and an Extended Text Table (ETT). The MGT provides information about all other *PSIP* tables with the exception of the System Time Table (not relevant here). *PSIP*, Section 6.2, p. 15. For example, the MGT

provides packet identifier information to the receiver so that the receiver can identify packets conveying particular types of tables such as the EIT and ETT.

In particular, an EIT is a separate type of *PSIP* table and "contains information (titles, start times, etc.) for events on defined virtual channels" over a 3 hour interval of time. There are at least four different EITs transmitted in *PSIP* for covering a twelve hour time interval and there can be as many as 128 different EIT tables. *PSIP*, p. 30.

Likewise, an ETT is also a separate type of table and coveys text, referred to as an Extended Text Message (ETM) (hereafter referred to simply as a "message"). A message can be of two types - a channel message or an event message. A channel message is used to provide detailed description of virtual channels and, in this case, the ETT is a "channel ETT". An event message is used to describe events and, in this case, the ETT is an "event ETT". *PSIP*, p. 33-34. It is important to note that each EIT can have one ETT. *PSIP*, p. 74. Further, an EIT is linked to a message in the ETT via a parameter event_id of the EIT. *PSIP*, p. 79.

Not only are the EIT and ETT separate tables in *PSIP*, but they are also transmitted separately. In other words, the EIT does not include the ETT and the ETT does not include the EIT. This is clearly shown in Tables 6.2 and 6.3 of *PSIP* illustrating the syntax for the MGT. Table 6.2 of *PSIP* for the MGT is shown below, with an indication of the relevant parts.

Table 6.2 Bit Stream Syntax for the Master Guide Table

Syntax	Bits	Format
master_guide_table_section () {		
table_id	8	0xC7
section_syntax_indicator	1	'1'
private_indicator	1	'1'
zero	2	'00'
section_length	12	uimsbf
table_id_extension	16	0x0000
reserved	2	'11'
version_number	5	uimsbf
current_next_indicator	1	'1'
section_number	8	0x00
last_section_number	8	0x00
protocol_version	8	uimsbf
tables_defined	16	uimsbf
for (i=0;i<tables_defined;i++) {		
table_type	16	uimsbf
reserved	3	'111'
table_type_PID	13	uimsbf
reserved	3	'111'
table_type_version_number	5	uimsbf
number_bytes	32	uimsbf
reserved	4	'1111'
table_type_descriptors_length	12	uimsbf
for (k=0;k<N;k++)		
descriptor()	var	
}		
reserved	4	'1111'
descriptors_length	12	uimsbf
for (i = 0;i< N;i++)		
descriptor()	var	
CRC_32	32	rpchof
}		

Table 6.3 → table_type

packet identifier → table_type_PID

As can be observed from Table 6.2 of *PSIP* for the MGT, each table is identified by its table type and has an associated packet identifier (table_type_PID) for identifying the packet that conveys the information for the particular table. *PSIP*, p. 18. This information is further shown in Table 6.3 of *PSIP* (shown below)

Table 6.3 Table Types

table_type	Meaning
0x0000	Terrestrial VCT with current_next_indicator=1
0x0001	Terrestrial VCT with current_next_indicator=0
0x0002	Cable VCT with current_next_indicator=1
0x0003	Cable VCT with current_next_indicator=0
0x0004	channel ETT
0x0005-0x00FF	[Reserved for future ATSC use]
0x0100-0x017F	EIT-0 to EIT-127
0x0180-0x01FF	[Reserved for future ATSC use]
0x0200-0x027F	event ETT-0 to event ETT-127
0x0280-0x0300	[Reserved for future ATSC use]
0x0301-0x03FF	RRT with rating_region 1-255
0x0400-0x0FFF	[User private]
0x1000-0xFFFF	[Reserved for future ATSC use]

ETTs → 0x0004, 0x0200-0x027F

EITs → 0x0100-0x017F

As can be observed from Table 6.3 of *PSIP*, there are separate definitions for a channel ETT, an event ETT and an EIT. Thus, ETTs or EITs are separately defined and separately

transmitted **in different packets** since each table will have its own "table_type_PID" as shown in Table 6.2 of *PSIP*.

The fact that an EIT is transmitted separate and apart from an ETT is also shown in Figures 5.1 of *PSIP*, reproduced below.

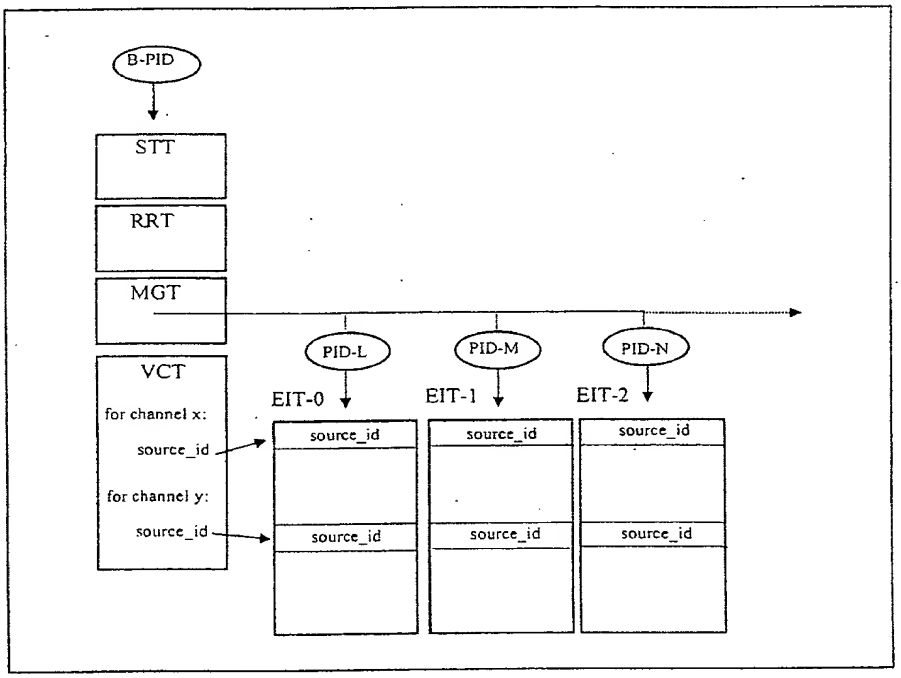


Figure 5.1 Table hierarchy for the Program and System Information Protocol (PSIP)

As can be observed from Figure 5.1 of *PSIP*, the MGT includes information that points to the packets that convey EITs (table_type_PID of Table 6.2).

Likewise, the fact that an ETT is transmitted separate and apart from an EIT is shown in Figures 5.2 of *PSIP*, reproduced below.

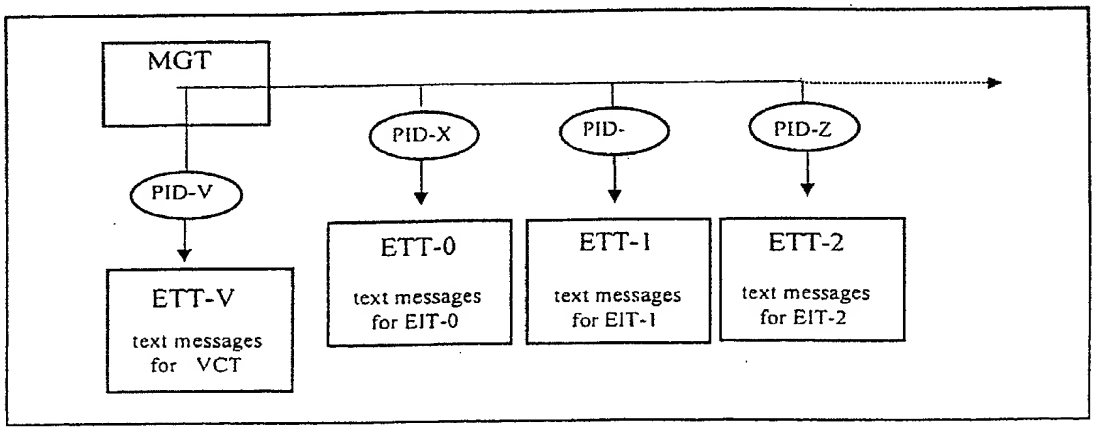


Figure 5.2 Extended Text Tables (ETTs) defined to carry text messages for describing virtual channels and events.

Again, as can be observed from Figure 5.2 of *PSIP*, the MGT includes information that points to the packets that convey ETTs (table_type_PID of Table 6.2).

Thus, any EIT tables and ETT tables are transmitted separately in *PSIP*.

The Examiner's Analysis of PSIP is Wrong

The Examiner asserts that an ETT is a tertiary table hierarchically lined to an EIT, a secondary table – this is incorrect. As described above, an EIT and an ETT are not only separate tables – but they are treated the same in *PSIP*. There is no ranking, or order as between the EIT and the ETT as required by Applicants' claim 1. Indeed, on its face, Figures 5.1 and 5.2 of *PSIP* (shown above) show that both the EIT and ETT are, at best, secondary tables, i.e., they are at the same level – not different levels as required by Applicants' claim 1. Further, reference to Table 6.3 of *PSIP* (shown above) shows that the EIT and ETT are treated that same – as just different types of tables.

Nor does that fact that an ETT can be associated with an EIT as shown in Figure 5.2 of *PSIP* (shown above) remedy this deficiency. Indeed, even though *PSIP* describes the fact that an EIT is linked to a message **in** the ETT (*PSIP*, p. 79) – Applicants' claim 1 requires more. In particular, Applicants' claim 1 **requires a hierarchical linking between the** ETT and the EIT **tables**. Yet, nowhere does *PSIP* show such a hierarchical linking between the EIT and the ETT tables. For example, the EIT includes no information about the ETT. Likewise, an ETT includes no information about the EIT. Indeed, it is important to note that an EIT *only includes information to locate a message* and does not include information to locate an ETT. In particular, the EIT includes the following information to locate a message – the ETM_location. An EIT structure as illustrated in Table 6.12 of *PSIP* is shown below.

Table 6.12 Bit Stream Syntax for the Event Information Table

Syntax	Bits	Format
event_information_table_section () {		
table_id	8	0xCB
section_syntax_indicator	1	'1'
private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
source_id	16	uimsbf
zero	2	'00'
version_number	5	uimsbf
current_next_indicator	1	'1'
section_number	8	uimsbf
last_section_number	8	uimsbf
protocol_version	8	uimsbf
num_events_in_section	8	uimsbf
for (j = 0; j< num_events_in_section;j++) {		
reserved	2	'11'
event_id	14	uimsbf
start_time	32	uimsbf
reserved	2	'11'
ETM_location	2	uimsbf
length_in_seconds	20	uimsbf
title_length	8	uimsbf
title_text()	var	
reserved	4	'1111'
descriptors_length	12	
for (i=0;i<N;i++) {		
descriptor()		
}		
}		
CRC_32	32	rpchof
}		

ETM location

Whether there is a message for an EIT - and where the message might be - is specified by the ETM_location parameter. The parameter has the values shown in Table 6.13 of *PSIP*, reproduced below.

Table 6.13 ETM_location

ETM_location	Meaning
0x00	No ETM
0x01	ETM located in the PTC carrying this PSIP
0x02	ETM located in the PTC carrying this event
0x03	[Reserved for future ATSC use]

As can be observed from Table 6.13 of *PSIP*, the ETM_location parameter specifies if there is no message or, if there is a message, in which physical transmission channel (PTC) the message might be found. Nowhere is there ETT information. Thus, there is no hierarchical linking between an EIT and an ETT as required by Applicants' claim 1.

Likewise, *Eyer* is also found lacking with respect to these requirements of Applicants' claim 1. In particular, *Eyer* merely describes the use of a single version number for a block of data. (*Eyer*, col. 13, lns. 37-42.) Nowhere does *Eyer* describe or suggest a

first version identifier, a second version identifier and a processor for determining a change as required by Applicants' claim 1.

Notwithstanding the above, nowhere does either *Eyer*, or *PSIP* – singly or in combination – describe or suggest Applicants' claim requirement of **a second version identifier conveyed in a secondary data table and updated in response to ... a version change in the tertiary table**.

Turning first to the EIT and ETT, each table has its own **separate** version_number. The definition for the EIT version number is provided below.

> version_number - This 5-bit field is the version number of EIT-i. The version number shall be incremented by 1 modulo 32 when any field in the EIT-i changes. Note that the version _number for EIT-i has no relation with that for EIT-j when j is not equal to i.
>
> *PSIP*, p. 31, emphasis added.

From the above definition the following should be noted. First, although there can be up to 128 EITs, the version number for each EIT is different. Finally, the EIT version number is only changed when a field in the EIT changes. **In other words, the EIT version number is not affected by a change in the version number of the ETT – but only affected by a change in a field in the EIT.**

Turning now to the definition for the ETT version number, this is provided below.

> version_number - For the channel ETT, this 5-bit field indicates the version number of the channel ETT. The version number shall be incremented by 1 modulo 32 when any ETM in the channel ETT changes. For event ETT, This 5-bit field indicates the version number event ETT-i, where i, as in the EIT case, is the index of time span. The version number shall be incremented by 1 modulo 32 when any ETM in the event ETT-i changes. Note that the version _number for ETT-i has no relation with that for ETT-j when j is not equal to i.
>
> *PSIP*, p. 34, emphasis added.

Again, from the above definition the following should be noted. First, the version number for each event ETT is different. Finally, the ETT version number is only changed when an event message changes. **In other words, the ETT version number is not affected by a change in the EIT – but only affected by a change in an event message.**

It is noted that the Examiner points to pp. 72-74 of *PSIP* as an example of Applicants' claimed requirement. Yet the fact is, nowhere does pp. 72-74 of *PSIP* describe a change in the version number of an ETT that causes the version number of an EIT to change

as claimed by Applicants. Applicants can only find a description on pp. 72-74 of *PSIP* with regard to an EIT version number change. There is no mention of an ETT version change.

In view of the above, Applicants' claim 1, and dependant claims 2-5, are patentable since neither *Eyer* or *PSIP*, single or in combination, describe or suggest the requirements of Applicants' claim 1.

CLAIMS 8-12 ARE PATENTABLE

Similar comments apply to Applicants' independent claim 8. In particular, Applicants' independent claim 8 requires at least

> (1) a tertiary table hierarchically linked to said secondary table; and
>
> (2) a second version identifier conveyed in a secondary data table and updated in response to ... a version change in the tertiary table.
>
> Claim 8, lns. 8-13; emphasis added.

As such, Applicants' claims 8-12 stand or fall with Applicants' independent claim 1.

CLAIMS 6-7, 13-16

CLAIMS 6-7 ARE PATENTABLE

Applicants' independent claim 6 requires at least

> (1) partition identifiers assigned to individual partitions of said program guide data; and
>
> (2) wherein said program guide data partitions are dynamically re-partitionable by re-assignment of said partition identifiers **in said partitioning information**.
>
> Claim 6, lns. 6-8; emphasis added.

Nowhere does either *Eyer*, or *PSIP* – singly or in combination – describe or suggest a partition identifiers as claimed by Applicants.

The Examiner appears to argue that "table_type_PID" (packet identifier or packet ID) found in the MGT is a partition identifier as claimed by Applicants. As a further example, the Examiner points to the shift of EIT tables after the expiration of a three hour interval. Respectfully, the Examiner is wrong. As can be observed from p. 73 of *PSIP*, the actual values for a PID do not change. Indeed, this portion of *PSIP* describes an operation **in the**

receiver. The fact that the receiver now associates the PID for EIT-1 as representing EIT-0 **has nothing to do** with Applicants' claimed program guide structure.

First, Applicants' independent claim 6 requires a program guide that includes a "partition identifier" and that partitions are dynamically re-partitionable **by re-assignment** of said partition identifiers in said partitioning information. This is not described or suggested in *PSIP*. Nowhere does *PSIP* describe changing the value of a packet identifier in the program guide. While a receiver is free to associate packet ID values as it sees fit (e.g., see p. 73 of *PSIP*), the fact is that **packet ID values themselves are not changed** because this is how the packets are identified by the receiver.

In view of the above, Applicants' independent claim 6, and dependant claim 7, are patentable since the combination of *Eyer* and *PSIP* does not yield the requirements of Applicants' claim 6.

CLAIMS 13-16 ARE PATENTABLE

Similar comments apply to Applicants' independent claim 13. In particular, independent claim 13 requires:

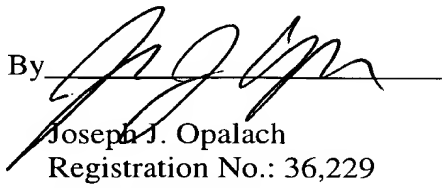
> cell numbers assigned to individual partitions of said program guide information, wherein said program guide information cell partitions are dynamically re-partitionable by re-assignment of said cell number in said database;
>
> Claim 13, lns. 8-10; emphasis added.

As such, Applicants' claims 13-16 stand or fall with Applicants' independent claim 6.

VIII. CONCLUSION

For the above reasons, Applicants submit that claims 1-16 are patentable over *Eyer* in view of *PSIP*. It is therefore respectfully requested that the rejection of claims 1-16 under 35 U.S.C. § 103(a) be reversed.

Respectfully submitted,
Mehmet Kemal Ozkan et al.

By 
Joseph J. Opalach
Registration No.: 36,229
(609) 734-6839

Patent Operations
Thomson Licensing LLC.
P.O. Box 5312
Princeton, New Jersey 08543-5312

March 2, 2007

IX. CLAIMS APPENDIX

1. (Original) Apparatus for acquiring packetized program data from at least a first source, comprising:

a processor for acquiring program guide information and for acquiring ancillary information conveyed in hierarchically ordered data tables in said packetized program data, said ancillary information including,

(a) a first version identifier conveyed in a primary data table and updated in response to a version change in at least one of a plurality of secondary tables hierarchically linked to said primary data table, and

(b) a second version identifier conveyed in a secondary data table and updated in response to at least one of,

a version change in said secondary table, and

a version change in a tertiary table hierarchically linked to said secondary table;

a processor for determining change in said secondary data table content by examining said second version identifier for a change following determination of a change in said first version identifier; and

an acquisition processor for acquiring said secondary data table in response to said determination of change.

2. (Original) Apparatus according to claim 1, wherein

said primary data table comprises a root database table for indicating version change in hierarchically ordered program guide data tables.

3. (Original) Apparatus according to claim 1, wherein

said secondary data table is used to indicate change in multimedia objects comprising objects associated with at least one of (a) broadcast channels, (b) broadcast programs, and (c) User interface controls.

4. (Original) Apparatus according to claim 1, wherein

said primary data table is used to indicate change in at least one of (a) electronic program guide information tables and (b) MPEG compatible program specific information.

5. (Original) Apparatus according to claim 1, wherein

said ancillary information is a two level hierarchical arrangement containing only a primary table and secondary tables.

6. (Original) Apparatus for adaptively decoding re-partitionable packetized program guide data, comprising:

a processor for acquiring program guide data comprising hierarchically ordered data table partitions and including partitioning information, said partitioning information including,

partition identifiers assigned to individual partitions of said program guide data, wherein said program guide data partitions are dynamically re-partitionable by re-assignment of said partition identifiers in said partitioning information; and

a processor for identifying said re-assigned partition identifiers and for acquiring additional program guide data in response to said identified re-assigned partition identifiers.

7. (Original) Apparatus according to claim 6, wherein

said partition identifiers identify program guide data partitions based on at least one of, (a) an area, (b) a broadcast time, (c) a complexity level, and (d) a partition type.

(claims continued on the next page)

8. (Original) A method for forming packetized program data to be suitable for processing in a decoder, comprising the steps of:

forming program guide information and ancillary information into hierarchically ordered data tables and including in said ancillary information,

(a) a first version identifier conveyed in a primary data table and updated in response to a version change in at least one of a plurality of secondary tables hierarchically linked to said primary data table, and

(b) a second version identifier conveyed in a secondary data table and updated in response to at least one of,

a version change in said secondary table, and

a version change in a tertiary table hierarchically linked to said secondary table; and

incorporating said ancillary information and said program guide information into packetized data for output to a transmission channel.

9. (Original) A method according to claim 8, including the step of

forming said primary data table to comprise a root database table for indicating version change in hierarchically ordered program guide data tables.

10. (Original) A method according to claim 8, wherein

forming said secondary data table to indicate change in multimedia objects comprising objects associated with at least one of (a) broadcast channels, (b) broadcast programs, and (c) User interface controls.

11. (Original) A method according to claim 8, wherein

forming said primary data table to indicate change in at least one of (a) electronic program guide information tables and (b) MPEG compatible program specific information.

12. (Original) A method according to claim 8, wherein

said ancillary information is a two level hierarchical arrangement containing only a primary table and secondary tables.

13. (Original) A method for forming packetized program data to be suitable for processing in a decoder, comprising the steps of:

partitioning program guide information and ancillary information into hierarchically ordered data table partitions and including a database in said ancillary information, said database including,

(a) updatable version numbers for indicating content change of a partition, and

(b) cell numbers assigned to individual partitions of said program guide information, wherein said program guide information cell partitions are dynamically re-partitionable by re-assignment of said cell number in said database; and

incorporating said ancillary information and said program guide information into packetized data for output to a transmission channel.

14. (Original) A method according to claim 13, wherein

said ancillary information contains a multimedia object comprising objects associated with at least one of (a) broadcast channels, (b) broadcast programs, and (c) User interface controls.

15. (Original) A method according to claim 14, wherein

an object comprises at least one of (a) a video segment, (b) an audio segment, (c) text, (d) an icon representing a user selectable item for display, (e) an HTML or SGML document (f) a menu of selectable items, (g) an image window for presentation within an encompassing image, and (h) an image window for initiating a multimedia function.

16. (Original) A method according to claim 13, wherein

a cell number incorporates at least one of, (a) an area identifier, (b) a broadcast time identifier, and (c) a complexity level identifier.

17. (Canceled).

18. (Canceled).

X. EVIDENCE APPENDIX (**NONE**)

None.

XI. RELATED PROCEEDINGS APPENDIX (NONE)

None.